



Universidade Federal do Ceará – UFC  
Centro de Ciências – CC  
Departamento de Computação - DC  
Fundamentos de Programação

Exercício: Refatoração de Código

Objetivos: Utilizar, em conjunto, os diversos recursos de uma linguagem de programação.

Data da Entrega: 04/07/2022 (Trabalho Opcional – Valendo 2 pontos na 2ª. Avaliação)

NOME: \_\_\_\_\_ MATRÍCULA: \_\_\_\_\_

### **QUESTÃO 1**

Para cada um dos problemas a seguir, implemente um programa utilizando a Linguagem C (para alunos do curso de EC) ou Python (para alunos do curso de CC).

1. Crie um programa chamado Banco, o qual deve simular **um banco que possui apenas três contas**. Cada conta será representada pelo seu **número** (que deve ser um valor inteiro), pelo seu **saldo** (que deve ser um valor float) e por um **tipo** (que deve ser uma cadeia de caracteres de tamanho 35). O programa inicia pedindo o número das três contas a serem criadas. Cada conta inicia com saldo igual a zero. O programa deverá ficar em *loop* lendo as possíveis operações que podem ser realizadas em uma conta (**crédito**, **débito**, **transferir** e **consultar saldo**), até ser lido o número 0 (*flag* de saída). Quando o usuário escolher a opção “creditar”, deve-se pedir o número da conta e o valor a ser creditado. O programa deve então atualizar o saldo da conta correspondente, caso o número da conta fornecido pelo usuário seja igual ao número de uma das três contas criadas inicialmente, ou seja, caso a conta exista. Caso o usuário escolha a opção “debitar”, deve-se pedir o número da conta e o valor a ser debitado. Em seguida, deve-se atualizar o saldo da conta correspondente. Quando o usuário escolher a opção “saldo”, deve pedir o número da conta e imprimir o saldo da mesma. Caso o usuário escolha a opção “transferir”, o programa deve ler do teclado o número da conta de origem, o número da conta de destino e o valor a transferir. Em seguida, deve atualizar os saldos da conta de origem e da conta de destino (de acordo com o valor a ser transferido). Ao encerrar o programa deve ser impresso o número e o saldo de cada uma das três contas. **Não utilize vetores, matrizes, registros, arquivos ou funções.**
2. A partir do programa anterior, crie um novo programa chamado BancoN capaz de armazenar e gerenciar “n” contas bancárias, onde “n” é um número fornecido pelo usuário. **Utilize vetores e ou matrizes, porém não use registros, arquivos ou funções.**
3. A partir do programa anterior, crie um novo programa chamado BancoFuncao capaz de armazenar e gerenciar “n” contas bancárias, onde “n” é um número fornecido pelo usuário. As contas devem ser armazenadas em um vetor. **Utilize funções** quando necessário. **Não utilize registros ou arquivos.**
4. A partir do programa anterior, crie um novo programa chamado BancoRegistro capaz de armazenar e gerenciar “n” contas bancárias, onde “n” é um número fornecido pelo usuário. **As contas devem ser representadas por registros** e armazenadas em um vetor.

5. Crie um programa chamado BancoTXT, o qual deve simular um banco, o qual poderá possuir diversas contas. Cada conta será representada pelo seu número, que deve ser uma cadeia de caracteres, e pelo seu saldo, que deve ser do tipo numérico. O programa deverá ficar em loop lendo as possíveis operações que podem ser realizadas em um banco (criar uma conta, remover uma conta, realizar crédito, realizar débito, realizar transferência de valores entre duas contas e consultar o saldo de uma determinada conta), até ser lido o número 0 (flag de saída). Quando o usuário escolher a opção “creditar”, deve-se pedir o número da conta e o valor a ser creditado. O programa deve então atualizar o saldo da conta correspondente. Caso não exista uma conta com o número fornecido, uma mensagem de erro deverá ser exibida. Caso o usuário escolha a opção “debitar”, deve-se pedir o número da conta e o valor a ser debitado. Em seguida, deve-se atualizar o saldo da conta correspondente. Caso não exista uma conta com o número fornecido, uma mensagem de erro deverá ser exibida. Quando o usuário escolher a opção “saldo”, deve pedir o número da conta e imprimir o saldo da mesma. Caso não exista uma conta com o número fornecido, uma mensagem de erro deverá ser exibida. Caso o usuário escolha a opção “transferir”, o programa deve ler do teclado o número da conta de origem, o número da conta de destino e o valor a transferir. Em seguida, deve atualizar os saldos da conta de origem e da conta de destino (de acordo com o valor a ser transferido). Caso a conta de origem ou a conta de destino não exista, uma mensagem de erro deverá ser exibida. Quando o usuário escolher a opção “remover”, deve-se pedir o número da conta a ser removida e em seguida excluir a conta. Caso não exista uma conta com o número fornecido, uma mensagem de erro deverá ser exibida. **As contas devem ser representadas por registros e armazenadas em um arquivo texto. Não utilize funções.**
6. A partir do programa anterior, crie um programa chamado BancoBIN onde as **contas são armazenadas em um arquivo binário. Não utilize funções.**
7. Refatore o programa chamado BancoTXT, a fim de organizar o código **utilizando sub-rotinas (funções).**
8. Refatore o programa chamado BancoBIN, a fim de organizar o código **utilizando sub-rotinas (funções).**

“Quando a educação não é libertadora,  
o sonho do oprimido é ser o opressor.”

**Paulo Freire**